IN THE CLAIMS

1. [currently amended] A process comprising:

~~for~~ automatically translating a formal language specification written in a formal language defining a full and complete computer program to be automatically written by a computer into a full and complete source code computer program that can be compiled into a complete, executable program which can execute by itself on a computer and needs no additional third party source code or source code from existing components or code libraries to be compiled with it to make said complete executable program and which implements the requirements of said formal language specification, said formal language specification defining at least classes of objects having attributes, services and relationships with other classes, ~~said specification written in a formal language,~~ comprising the following steps:

using a computer, automatically write computer code that will request user name and password, receive any responses and authenticate the user;

using a computer, automatically write computer code that will determine this user's privilege level and query said formal language specification and determine all object attributes this user has privilege to see and query and all services this user can invoke;

using a computer, automatically write computer code which queries said formal language specification for all services of all classes that any authorized user may invoke and identifies an object server which will implement each said service;

using a computer, automatically write code that will retrieve service arguments for all services from one or more of a user, an ~~or from another~~ object server, and ~~or from~~ another process, as appropriate;

using a computer, automatically write code that controls a computer to display ~~displays menus options, icons or creates any other~~ means by which and entity ~~a user or another process~~ can invoke a service, and which receives input to invoke a particular service and responds by sending a message to the appropriate object server to invoke the service, said message including the necessary arguments for the service to execute;

using a computer, automatically write code that implements an object server for every service, each of which first checks to verify that state transitions are valid and make sense for the current state of objects of which the object service will be altering the state~~of~~;

using a computer, automatically write code for every object server that verifies preconditions are satisfied before making state transitions of any objects the states of which are acted upon by the object server;

using a computer, automatically write code to make all valuation calculations required by said formal language specification of each object server;

using a computer, automatically write code to verify that integrity constraints specified in said formal language specification on the values of attributes of objects have been satisfied after execution of a service and take action if said integrity constraints are not satisfied; and

using a computer, automatically write code for every object server to test trigger relationships specified in said formal language specification after execution of a service and carry out appropriate action if a trigger event has occurred.

2. [currently amended] An apparatus for automatically translating a formal

specification written in a formal language which has predefined rules of grammar, said formal specification defining a computer program to be automatically written by a computer, said translating acting to convert said formal specification into a computer program that implements the requirements of said formal specification, said formal specification defining at least classes of objects having attributes, services and relationships with other classes, ~~said specification written in a formal language,~~ comprising:

a computer programmed with an operating system and one or more other programs to cooperate with said operating system to control said computer to perform the following functions:

automatically write computer code that will request user name and password, receive any responses and authenticate the user;

automatically write computer code that will determine a ~~this~~ user's privilege level and query said formal ~~language~~ specification and determine all object attributes said ~~this~~ user has privilege to see and query and all services said ~~this~~ user can invoke;

automatically write computer code which queries said formal specification for all services of all classes that any authorized user may invoke and identifies an object server which will implement said service;

automatically write code that will retrieve service arguments for all services from one or more of a user, ~~or from~~ another object server, and ~~or from~~ another process, as appropriate;

automatically write code that displays one or more user interface tools ~~menus options, icons or creates any other means by which a user or another process can invoke~~ which can be used to invoke a service, and which receives input to invoke a particular service and which responds by sending a message to the appropriate object server to invoke said ~~the~~ service, said message including the necessary arguments for said ~~the~~ service to execute;

automatically write code that implements an object server for every service, each of which first checks to verify that state transitions are valid and make sense for the current state of objects the object service will be altering the state of;

automatically write code for every object server that verifies preconditions are satisfied before making state transitions of any objects the states of which are acted upon by said ~~the~~ object server;

automatically write code to make all valuation calculations required by said specification of each object server;

automatically write code to verify that integrity constraints specified in said formal specification on the values of attributes of objects have been satisfied after execution of a service and take action if said integrity constraints are not satisfied; and

automatically write code for every object server to test trigger relationships specified in said formal specification after execution of a service and carry out appropriate action if a trigger event has occurred.

3. [currently amended] A computer-readable medium containing instructions for controlling a computer to automatically translate a formal specification defining a computer program to be automatically written by a computer into a computer program that implements the requirements of said formal specification, said formal specification defining at least classes of objects having attributes, services and relationships with other classes, said formal specification written in a formal language having predefined rules of grammar, by:

automatically writing computer code that will request user name and password, receive any responses and authenticate the user;

automatically writing computer code that will determine a ~~this~~ user's privilege level and query said formal ~~language~~ specification and determine all object attributes said ~~this~~ user has privilege to see and all services said ~~this~~ user can invoke;

automatically writing computer code which queries said formal specification for all services of all classes that any authorized user may invoke and identifies an object server which will implement said service;

automatically writing computer code that will retrieve service arguments for all services from one or more of a user, an ~~or from another~~ object server, and ~~or from~~ another process, as appropriate;

automatically write code that displays menus options, icons or creates any other means by which a user or another process can invoke a service, and which receives input to invoke a particular service and responds by sending a message to the appropriate object server to invoke the service, said message including the necessary arguments for the service to execute;

automatically writing code that implements an object server for every service, each of which first checks to verify that state transitions are valid and make sense for the current state of objects the object service will be altering the state of;

automatically write code for every object server that verifies preconditions are satisfied before making state transitions of any objects the states of which are acted upon by the object server;

automatically write code to make all valuation calculations required by said formal specification of each object server;

automatically write code to verify that integrity constraints specified in said formal specification on the values of attributes of objects have been satisfied after execution of a service and take action if said integrity constraints are not satisfied; and

automatically write code for every object server to test trigger relationships specified in said formal specification after execution of a service and carry out appropriate action if a trigger event has occurred.